Network Anomaly Detection

Team Members:
Ken Lei, Nathan Anderson

Product Overview

- **Project domain:** Network security
- Network intrusion detection
  - Identifies potentially malicious activities on a network
  - Alerts administrators of suspicious behavior
  - Trade-off between precision and recall
    - We want to detect unusual activity
    - But… overloading the administrator is counter-productive
- We used unsupervised machine learning to monitor network flows and warn NetOps about possible attacks in real time

Product AI Canvas

Opportunity
Why do it?

Companies lose money due to unauthorized activities on their networks.

Solution
What is it?

Autoencoder to detect and report anomalous behaviors on a network

Consumers
Who needs it?

Companies with significant networks that must be reliable.

Data
What are the model's inputs?

Session metadata, such as Flow Bytes/s, Flow Packets/s

Strategy
Why us?

Novel ML approach to this problem.

Policy & Process
What else must change?

Administrator(s) need to monitor resulting NetOps alerts.

Transfer Learning
How will we build it?

Existing flow-based network anomaly detection researches.

Success Criteria
How will we know it works?

Precision-Recall curve

Product Team

MODEL LEAD
-Ken-

- Store dataset
- Create data-processing pipeline
- Build unsupervised ML model

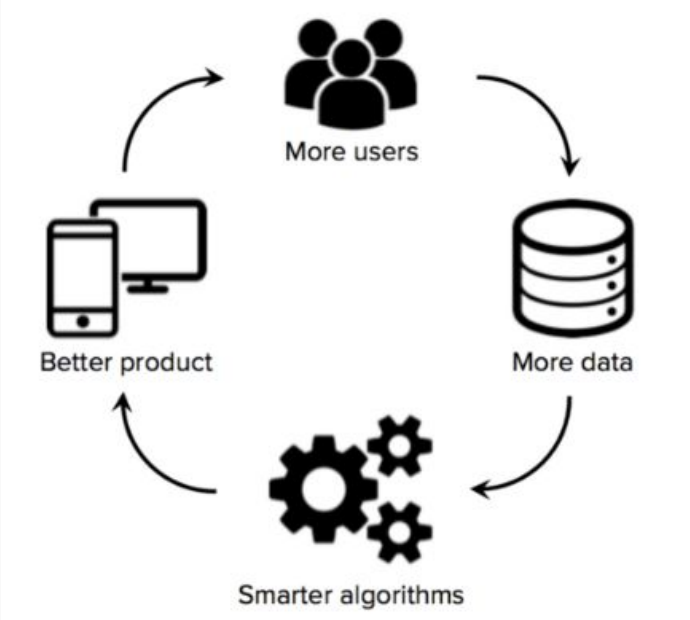
PRODUCT LEAD
-Nathan-

- Design performance metrics
- Monitor drift
- Deploy MVP

Product Value

- We generate business value by reducing costs associated with unauthorized network uses
- The exact value depends on how critical/expensive the network is for the business
- Potential to leverage data flywheel effects
  - As the model is used, new data is continuously collected
  - The additional data should improve the model
  - Making the business more likely to use the model more heavily
  - etc.

Data Flywheel



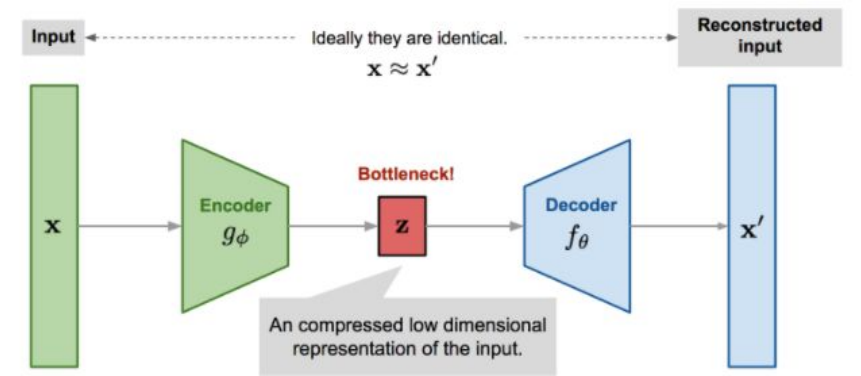
https://www.cbinsights.com/research/team-blog/data-network-effects/

Data

- Training data: CICIDS2017.
  - Most comprehensive dataset in this domain.
- Packet information is extracted using software that runs on the users' machines.
- It is then aggregated by session using tools like CICFlowMeter, which means that no sensitive data (packet payload) will be collected.
- The processed data is transmitted to a web storage platform and appropriately preprocessed for the model.

https://www.cbinsights.com/research/team-blog/data-network-effects/

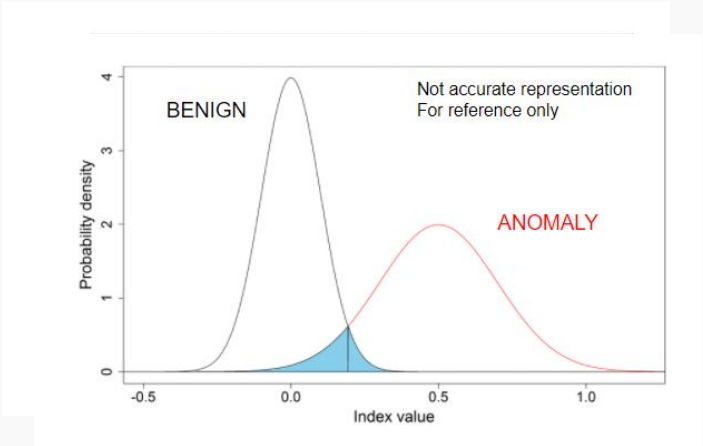
Model Selection

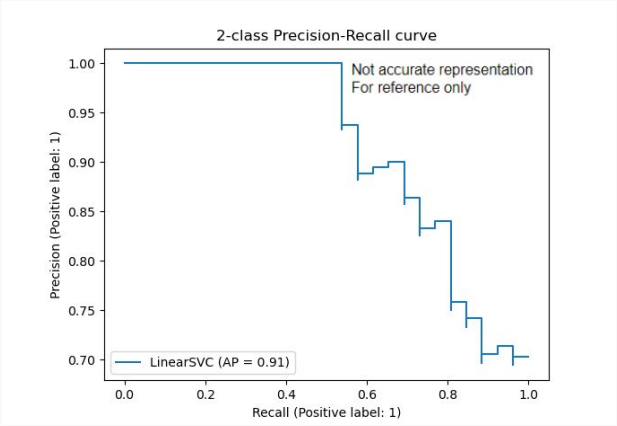


- Main Logic:
  If trained with BENIGN traffic, Autoencoders should be able to reproduce the input with high fidelity as long as the input is BENIGN traffic.

https://lilianweng.github.io/posts/2018-08-12-vae/

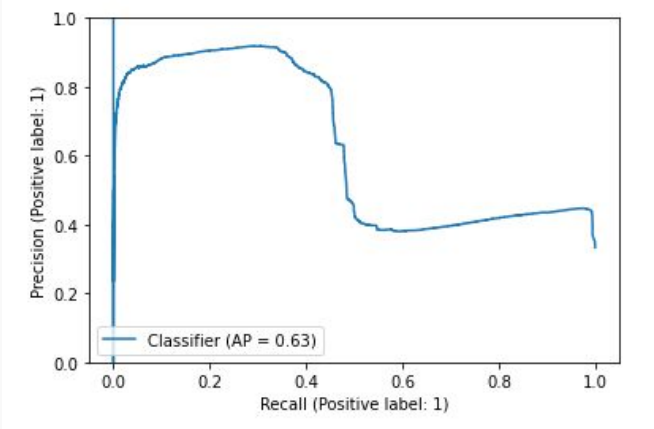
Evaluation Metrics





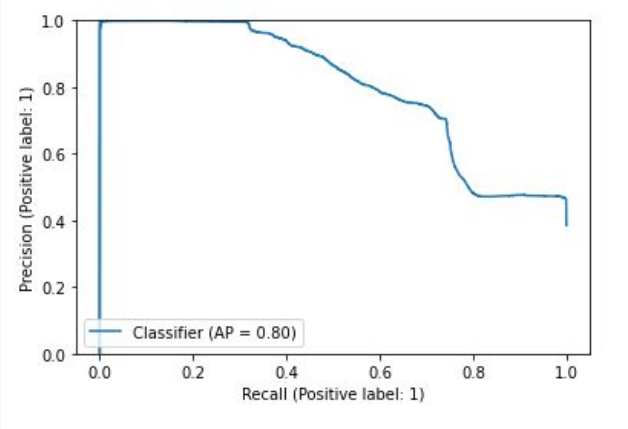
- Anomaly scores for BENIGN and ANOMALY will likely to overlap.
- Use “Average Precision” (AP) to the weighted mean of precisions achieved at each threshold.

Baseline Comparison



Heuristic Baseline

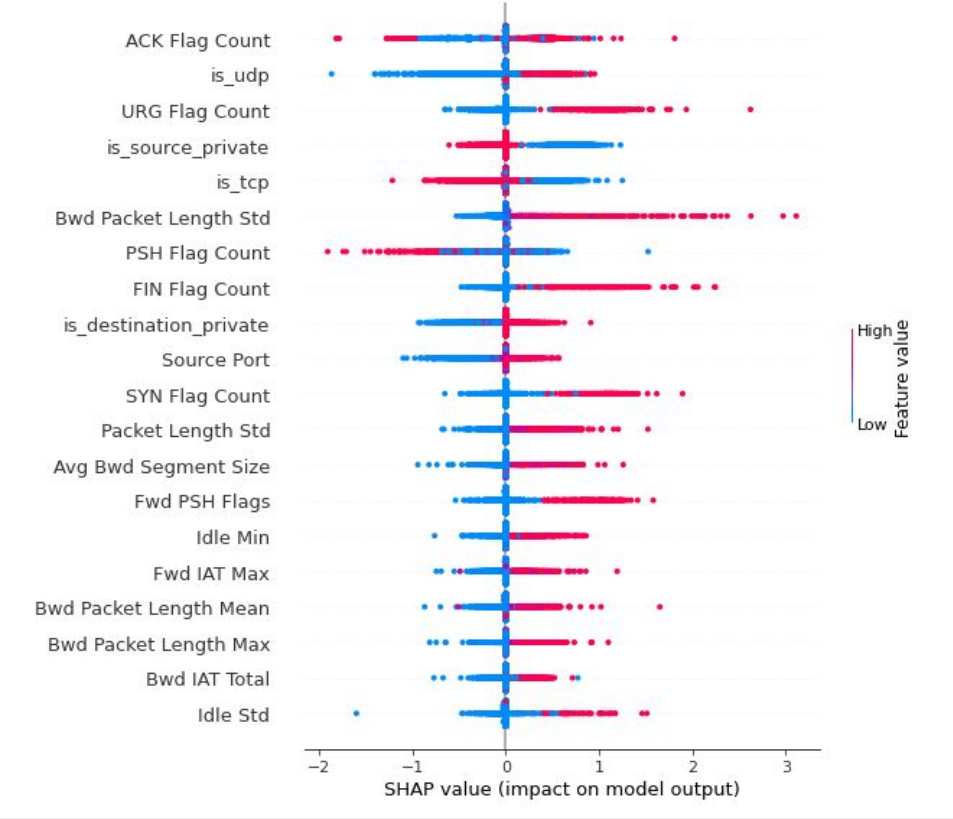
Determine Anomaly from standard deviation



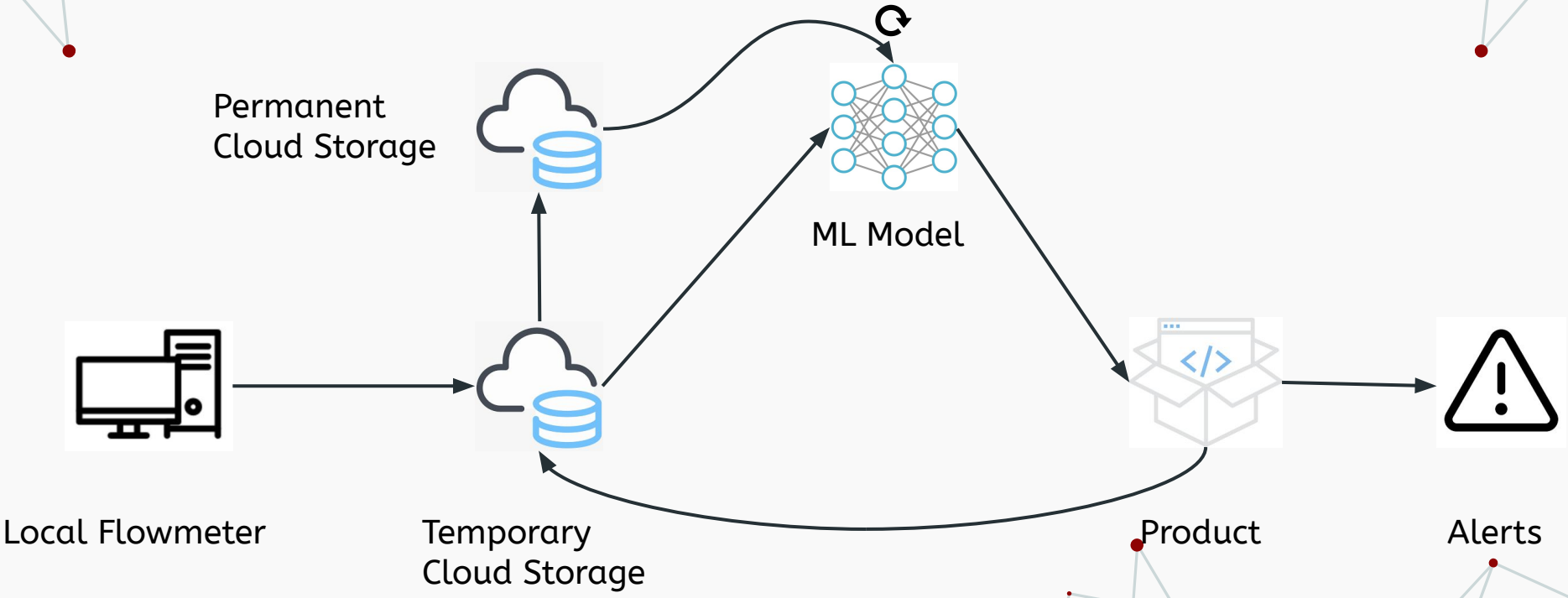
Our Model

Determine Anomaly from Reconstruction Loss

SHAP Values


ACK Flag Count
is_udp
URG Flag Count
is_source_private
is_tcp
Bwd Packet Length Std
PSH Flag Count
FIN Flag Count
is_destination_private
Source Port
SYN Flag Count
Packet Length Std
Avg Bwd Segment Size
Fwd PSH Flags
Idle Min
Fwd IAT Max
Bwd Packet Length Mean
Bwd Packet Length Max
Bwd IAT Total
Idle Std
−2
−1
0
1
2
3
SHAP value (impact on model output)
High
Low
Feature value

System Architecture



System Components

- FlowMeter:
  - Collect aggregated flow stats from the network deployed.
  - No sensitive data (packet payload) will be transmitted.

- Temporary Cloud Storage:
  - Temporary storage of data before being merged into permanent cloud storage.

- Permanent Cloud Storage:
  - Models can be re-trained with data stored here.
  - Performance improvement for global users or a specific user.

System Components

- Model:
  - Preferably runned on cloud computing platforms, e.g. EC2.
  - Currently on my local instance the testing time is 5 seconds for 500,000 flows. Increasing testing batch size or utilizing distributed computing will make it faster.
  - Need constant performance monitoring on the model. (preferably with DVC)

- Product:
  - Allow the user to choose the desired precision/recall level.
  - Alert the user if anomaly flow is detected.
  - Should be able to record user feedback, e.g. false positives.
  - Correct the label before data is being merged to permanent cloud storage.

MVP Demo

- App displays the current precision-recall curve
- Users can choose the desired precision/recall level
- App displays the chosen threshold and expected precision/recall (calculated based on test set)
- After user confirmation on the reported metric, the product reads several flows and generates predictions
- The product alerts the user if anomaly flow is detected, along with information of the flow, e.g. IP address, timestamp

- Link to MVP demo: https://share.streamlit.io/eurobait/project_demo/main/demo.py

THANKS

CREDITS: This presentation template was created by Slidesgo, including icons by **Flaticon**, infographics & images by **Freepik**